

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине

‘Распределённые системы хранения данных’

Вариант №536

Выполнил:

Студент группы Р33312

Соболев Иван

Александрович

Преподаватель:

Николаев Владимир

Вячеславович



УНИВЕРСИТЕТ ИТМО

Санкт-Петербург, 2024

Оглавление

Задание:	2
Выполнение:	2
Выводы по работе:	8

Задание:

Введите вариант: 536

Используя сведения из системных каталогов, получить информацию обо всех таблицах, на которые текущий пользователь может выдать права доступа другому заданному пользователю. Полученную информацию представить в виде списка следующего формата:

```
Текущий пользователь: s100000
Кому выдаём права доступа: s111111

No.  Имя таблицы
-----
1  TABLE1
2  TABLE2
3  TABLE3
...

```

Программу оформить в виде анонимного блока.

Выполнение:

Проанализировав доступные системные каталоги, я пришел к выводу, что необходимая информация хранится в каталогах - information_schema.role_table_grants, pg_tables.

Код анонимного блока:

DO

\$\$

DECLARE

row record;

from_user text := current_setting('s336760.current_user');

to_user text := current_setting('s336760.target_user');

table_num int := 1;

from_user_exists boolean;

to_user_exists boolean;

BEGIN

RAISE INFO 'Текущий пользователь: %', from_user;

RAISE INFO 'Кому выдаём права доступа: %', to_user;

RAISE INFO 'No. Имя таблицы';

RAISE INFO '--- -----';

```

FOR row IN
    SELECT DISTINCT table_name as name_of_table
    FROM information_schema.role_table_grants rg
        JOIN pg_tables pt ON rg.table_schema = pt.schemaname AND rg.table_name =
pt.tablename
    WHERE grantor = from_user
        and is_grantable = 'YES'
    ORDER BY name_of_table
LOOP
    RAISE INFO '% %', table_num, row.name_of_table;

    table_num := table_num + 1;
END LOOP;

END

$$;

```

Вывод:

```

INFO: Текущий пользователь: s100000
INFO: Кому выдаём права доступа: s111111
INFO: No.   Имя таблицы
INFO: ---   -----

```

Имеем пустой вывод так, как пользователя s100000 в нашей базе нет:

nasty		{
nikolaevv		{
ninomt		{
nnaumova		{
ozhiganov		{
pasha		{
platunov		{
polyakov		{
postgres	Суперпользователь, Создаёт роли, Создаёт БД, Репликация, Пропускать RLS	{
pvalakshin		{
roggiared		{
s119391		{
s125520		{
s126685		{
s134051		{
s134057		{
s141815		{
s141821		{
s146963		{
s148787		{
s154966		{

Попробуем поменять его на доступного s119391:

```

END $$,
INFO: Текущий пользователь: s119391
INFO: Кому выдаём права доступа: s111111
INFO: No. Имя таблицы
INFO: --- -----

```

Также имеем пустой вывод. Это связано с тем, что к базе я подключаюсь со своим пользователем, который не имеет права на просмотр таблиц и данных других пользователей. Попробуем заменить на мой номер s336760 и убедимся в работоспособности программы:

```

INFO: Текущий пользователь: s336760
INFO: Кому выдаём права доступа: s111111
INFO: No. Имя таблицы
INFO: --- -----
INFO: 1 administrator
INFO: 2 callcenterworker
INFO: 3 car
INFO: 4 detail
INFO: 5 detailprovider
INFO: 6 detailprovider_detail
INFO: 7 employee
INFO: 8 employee_sto
INFO: 9 images
INFO: 10 manufacturer
INFO: 11 offer
INFO: 12 offer_detail
INFO: 13 offer_purchase
INFO: 14 offer_tool
INFO: 15 purchase
INFO: 16 question
INFO: 17 review
INFO: 18 review_offer
INFO: 19 sto
INFO: 20 swaphub_images
INFO: 21 swaphub_products
INFO: 22 swaphub_user_role
INFO: 23 swaphub_users
INFO: 24 tool
INFO: 25 user_car
INFO: 26 user_role
INFO: 27 users
INFO: 28 users_questions
INFO: 29 worker
INFO: 30 worker_worker_requests
INFO: 31 workerrequest

```

Следует доработать блок так, чтобы выводились сообщения о том, что вводимых пользователей нет.

Код доработанного блока:

DO

\$\$

DECLARE

row record;

from_user text := current_setting('s336760.current_user');

to_user text := current_setting('s336760.target_user');

table_num int := 1;

from_user_exists boolean;

to_user_exists boolean;

```

BEGIN

    SELECT EXISTS(SELECT *
                  FROM pg_user
                  WHERE username = from_user)
    INTO from_user_exists;

    SELECT EXISTS(SELECT *
                  FROM pg_user
                  WHERE username = to_user)
    INTO to_user_exists;

    IF NOT from_user_exists THEN
        RAISE INFO 'Пользователь % на выдачу прав не найден', from_user;
    END IF;

    IF NOT to_user_exists THEN
        RAISE INFO 'Пользователь % на получение прав не найден', to_user;
        RETURN;
    END IF;


    RAISE INFO 'Текущий пользователь: %', from_user;
    RAISE INFO 'Кому выдаём права доступа: %', to_user;
    RAISE INFO 'No. Имя таблицы';
    RAISE INFO '--- -----';

    FOR row IN
        SELECT DISTINCT table_name as name_of_table
        FROM information_schema.role_table_grants rg
            JOIN pg_tables pt ON rg.table_schema = pt.schemaname AND rg.table_name =
pt.tablename

        WHERE grantor = from_user
            and is_grantable = 'YES'
        ORDER BY name_of_table
    LOOP

        RAISE INFO '% %', table_num, row.name_of_table;

        table_num := table_num + 1;
    
```

END LOOP;

END

\$\$;

Вывод:

```
INFO: Пользователь s100000 на выдачу прав не найден
INFO: Пользователь s111111 на получение прав не найден
```

Однако менять пользователей внутри кода неудобно, автоматизируем:

```
\echo 'Введите текущего пользователя: '
```

```
\prompt " current_user_name
```

```
select set_config('s336760.current_user', :'current_user_name', false);
```

```
\echo 'Введите имя пользователя, кому нужно выдать права доступа: '
```

```
\prompt 'Введите имя пользователя, кому нужно выдать права доступа: ' target_user_name
```

```
select set_config('s336760.target_user', :'target_user_name', false);
```

DO

\$\$

DECLARE

row record;

from_user text := current_setting('s336760.current_user');

to_user text := current_setting('s336760.target_user');

table_num int := 1;

from_user_exists boolean;

to_user_exists boolean;

BEGIN

SELECT EXISTS(SELECT *

FROM pg_user

WHERE username = from_user)

INTO from_user_exists;

SELECT EXISTS(SELECT *

FROM pg_user

WHERE username = to_user)

```

INTO to_user_exists;
IF NOT from_user_exists THEN
    RAISE INFO 'Пользователь % на выдачу прав не найден', from_user;
END IF;

IF NOT to_user_exists THEN
    RAISE INFO 'Пользователь % на получение прав не найден', to_user;
    RETURN;
END IF;

RAISE INFO 'Текущий пользователь: %', from_user;
RAISE INFO 'Кому выдаём права доступа: %', to_user;
RAISE INFO 'No. Имя таблицы';
RAISE INFO '--- -----';

FOR row IN
    SELECT DISTINCT table_name as name_of_table
    FROM information_schema.role_table_grants rg
        JOIN pg_tables pt ON rg.table_schema = pt.schemaname AND rg.table_name =
pt.tablename
    WHERE grantor = from_user
        and is_grantable = 'YES'
    ORDER BY name_of_table
LOOP
    RAISE INFO '% %', table_num, row.name_of_table;
    table_num := table_num + 1;
END LOOP;

END
$$;

```

Далее запускаем наш скрипт командой:

```
psql -h pg -d studs -f ~/script.sql 2>&1 | sed 's|.*INFO: ||g'
```

и имеем вывод:

```
[s336760@helios ~]$ psql -h pg -d studs -f ~/script.sql 2>&1 | sed 's|.*INFO: ||g'
Введите текущего пользователя:
s336760
Введите имя пользователя, кому нужно выдать права доступа:
s119391
Текущий пользователь: s336760
Кому выдаём права доступа: s119391
No.  Имя таблицы
---  -----
1 administrator
2 callcenterworker
3 car
4 cats
5 detail
6 detailprovider
7 detailprovider_detail
8 employee
9 employee_sto
10 images
11 manufacturer
12 offer
13 offer_detail
14 offer_purchase
15 offer_tool
16 pointz
17 purchase
18 question
19 review
20 review_offer
21 sto
22 swaphub_images
23 swaphub_products
24 swaphub_user_role
25 swaphub_users
26 tool
27 user_car
28 user_role
29 users
30 users_questions
31 uzers
32 worker
33 worker_worker_requests
34 workerrequest
```

Выводы по работе:

В ходе выполнения данной лабораторной работы я познакомился со внутренним устройством PostgreSQL, повторил `plpgsql`. Познакомился с анонимными блоками и на практике с помощью АБ узнал нужную мне информацию о таблицах.